



05

Instancias parte 1

Transcripción

[00:00] Ya tenemos nuestra clase cuenta creada y vamos a volver a nuestra presentación. Ya sabemos que cuenta es la abstracción de cierto grupo de campos que van a representar una entidad propiamente dicha. Ahora, ¿cómo es que yo comienzo digamos a llenar información dentro de estos campos?

[00:23] ¿Cómo es que yo comienzo a crear cuentas propiamente dicho, cuentas cada una con cantidad de dinero aún que tienen guardada, con su número de agencia, cada cuenta con su propio número? Existe un concepto llamado instancias. Las instancias digamos, son ya las representaciones propiamente dichas de ese objeto que va a ser la cuenta, pero ya representando en sí un objeto en particular.

[00:54] ¿Qué quiero decir con esto? Una instancia de cuenta va a utilizar este mismo esqueleto, esta misma estructura de datos, para representar la cuentaDeDiego en este caso. Tiene su propio número, tiene su propia agencia y tiene su propia cantidad de saldo.

[01:14] De la misma forma otra entidad, otro objeto cuenta también va a utilizar este mismo esqueleto creado aquí, representando los datos de cuenta de Alura por ejemplo, con su propio número de cuenta, su propio número de agencia y su propio saldo. Entonces, vamos nuevamente al código. Para eso vamos a crear nuevamente otra clase y vamos solamente a new class y como lo que vamos a hacer es crear una cuenta. Vamos a darle aquí crear cuenta.

[01:49] Le damos a finish y tenemos nuestra clase lista. Recordando un poco del curso anterior, vamos a escribir main "Ctrl + espacio", apretamos enter y tenemos nuestro método public static void main y lo que sigue ahí y vamos a crear una nueva cuenta. Ahora, ¿cómo yo le digo a Java "Java, créame una nueva cuenta basada en este archivo de aquí, porque representa lo que yo entiendo como cuenta con esos atributos, quiero que hagas eso por mí"? ¿Cómo sería?

[02:23] Java tiene una palabra reservada llamada new. Con new nosotros le estamos diciendo: "Crea esto nuevo". Entonces, ¿qué deseamos crear que sea nuevo? Una cuenta, punto y coma, y vemos que no está compilando. Ahora, vamos por partes. Vamos a usar como ejemplo nuestra cuenta aquí.

[02:49] Cuando se trata de atributos digamos, de campos, que van a contener un tipo de dato, vemos que él compila sin problemas, solo con punto y coma, pero en el caso de un objeto, cuando él representa la agrupación de uno o más campos, él ya no puede ser declarado de esta forma. Java nos exige que él sea declarado con paréntesis. ¿Por qué? Porque cuenta no es solamente un repositorio de datos.

[03:26] Cuenta es la agrupación de varios tipos de datos, varios datos, es un conjunto de datos que entre todos ellos representa una cuenta. Entonces, él nos pide que lo declaremos con paréntesis. ¿Es esto suficiente para crear una nueva cuenta? Es suficiente, sí, aquí ya estamos diciéndole: "Crea una nueva cuenta". Guardamos y ejecutamos, clic derecho, run as, Java application y vemos que no sucedió nada. En efecto no estamos haciendo nada aún.

[04:00] No le hemos dicho que imprima nada. Ahora, ya que tenemos nuestra cuenta creada y nosotros vamos a comenzar a insertar datos dentro de nuestro objeto cuenta, ¿de qué forma puedo yo acceder a esta cuenta que yo acabo de crear? ¿Cuál va a ser mi mecanismo para acceder a ese objeto y decirle "guarda tanta cantidad de saldo en el atributo saldo de esta cuenta"?

[04:27] La forma en la que yo puedo acceder a él es asignarle esta nueva cuenta a una variable, de la misma forma que lo hemos hecho digamos con nuestros campos simples, asignándole valores a las variables, este sería nuestro valor y le vamos a asignar una variable. Vamos a llamarla primera cuenta.

[04:52] Primera cuenta igual new cuenta. Pero nuevamente, como todo en Java es fuertemente tipado, aquí nos está exigiendo que le digamos qué tipo de variable es esta variable, esta instancia, qué tipo de objeto es. Es un objeto del tipo cuenta, ¿cierto? Porque estamos representando una cuenta. Entonces, aquí escribimos cuenta.

[05:22] Ahora quizás ustedes se están preguntando: "Pero Diego, yo ya le dije aquí que es una cuenta. ¿Por qué necesito escribir aquí que es cuenta?" O viceversa: "Yo ya estoy escribiendo aquí que es un objeto tipo cuenta. ¿Por qué yo necesito escribir aquí nuevamente? No tiene el menor sentido para mí". Espera. Esto de aquí por el momento es igual, pero van a ver casos en el futuro en el que no necesariamente este objeto que está aquí sea el objeto que está aquí.

[05:54] Ya cuando empecemos a trabajar con polimorfismo se van a dar cuenta que en realidad este tipo de sintaxis tiene muchas ventajas. Por el momento vamos a mantenerlo de esta forma. Entonces, vamos a repasar nuevamente. Yo estoy diciéndole que esta variable del tipo cuenta va a representar una nueva cuenta. Entonces, si yo deseo alterar el valor saldo de este objeto cuenta, ¿cómo podría yo acceder a este campo?

[06:40] Primero primeraCuenta, y si hacemos "Ctrl + espacio", automáticamente ya Eclipse nos va a sugerir con qué podemos autocompletar. En este caso yo voy a elegir primeraCuenta. Y si le doy punto, automáticamente vemos que él aquí me está listando agencia, número, saldo, titular, que casualmente son estos cuatro campos que yo especifiqué aquí.

[07:07] Entonces aquí Eclipse ya está entendiendo el objeto al que yo estoy haciendo referencia es un objeto del tipo cuenta y él me está preguntando aquí: "Diego, ¿tú con cuál de tus campos quieres trabajar?" Entonces yo voy a escoger saldo. Escojo saldo y le voy a decir que le asigne digamos 1000 de saldo.

[07:34] El código compila, perfecto. ¿Será que de verdad él guardó 1000 dentro del atributo saldo de primera cuenta? Vamos a comprobarlo. Entonces, vamos a darle un `system.out.print`. Vamos a imprimir `primeraCuenta`, "Ctrl + espacio", `primeraCuenta`, punto, `saldo`. Guardamos. Ejecutamos.

[08:02] Y en efecto puedo aumentar mi terminal, en efecto vemos que él guardó el valor de 1000 dentro del campo saldo de mi nuevo objeto cuenta.